REMARKS

This Amendment is in response to the Final Office Action dated April 20, 2005, and is requested to be entered in conjunction with entry of an RCE for the present application filed herewith. Applicant further requests to withdraw the pending notice of appeal.

In the Office Action, the Examiner rejected claims 1-7, 9, 17-19, 21-24, and 26-29 under 35 U.S.C. § 102(a) as being anticipated by Ha, U.S. Patent No. 6,175,919 (hereinafter *Ha*). Claims 8, 10, 13, 14, 25, and 30 were rejected under 35 U.S.C. § 103(a) as being unpatentable over *Ha*, in view of Rasmussen, U.S. Patent No. 6,640,334.

Independent claims 1, 13, 17, 22, and 26 are amended as shown above. Claims 1-30 remain pending in the application. For the reasons set forth below, the Applicants respectfully request reconsideration and allowance of all pending claims.

Argument in Support of Patentability of Amended Claims

In the Response to Arguments section of the present Office Action, the Examiner identifies that the previous versions of the claims did not recite any elements indicating that the first and second portions of firmware were loaded and executed during the ***same*** pre-boot phase. This was argued by the Applicant as a distinguishing feature over the Ha reference in the response filed on January 10, 2005.

During a telephone interview conducted August 18, 2005, the undersigned attorney and the Examiner discussed the issue of whether the previous state of the claims articulated that the aforementioned operations occurred during the same pre-boot phase. Applicant's position is that this was implied by the overall language of each of the independent claims, while the Examiner's position was that it was not in view of the broadest interpretation of the claims. To further the speed of prosecution, Applicant has agreed to explicitly provide such a claim limitation in the

independent claims. Accordingly, such language has been added to each of amended independent claims 1, 13, 17, 22, and 26.

For example, the amendment to claim 1 is illustrative of the amendments made to the independent claims, and now recites,

1. A method comprising:

loading platform firmware during a single pre-boot phase of a computer system by,

executing a first portion of platform firmware code that is stored locally in the computer system during the single pre-boot phase;

retrieving a second portion of platform firmware code from a remote firmware storage device; and

executing the second portion of platform firmware code during the single pre-boot phase. (Emphasis added)

Applicant respectfully asserts that *Ha* does not anticipate amended claim 1, or any of the other independent claims. In particular, with respect to claim 1, *Ha* does not retrieve a second portion of platform firmware code from a remote firmware storage device and execute that portion of platform firmware code during the same pre-boot phase of a computer system during which execution of a first portion of platform firmware code that is stored locally in the computer system is also performed. (Note: In accordance with the amended claims, the use of the terminology "single pre-boot phase" implies that the operations are performed during the same pre-boot phase.)

Ha discloses a method and apparatus for upgrading BIOS using serial communication. In support of the rejection of claims 1, 9, 17, 22, 24, 26 and 29, the Examiner cites the Summary, column 3 lines 45-65, column 4 lines 5-20 and 40-55, column 5 lines 5-25, and claims 1-3.

Claims 1-3 and column 3 lines 45-65 pertain to the embodiments shown in Figs 5 and 6. As recited in the *Ha* specification,

> The above-mentioned method for upgrading the BIOS will be described in detail with reference to FIGS. 5 and 6.
>
> FIG. 5 is a flow chart showing a process of transferring the BIOS upgrade software and BIOS image executed in the host computer in a method for upgrading the BIOS using serial communication according to the present invention.
>
> Referring to FIG. 5, in the process of transferring the BIOS upgrade software and BIOS image executed in the host computer (HOST), in step S50, it is determined whether the BIOS is requested to be upgraded by the personal computer. If not, a continuous check is made as to whether the BIOS is requested to be upgraded. When it is determined that the BIOS is requested to be upgraded during step S50, a model ID is obtained from the personal computer (step S51).
>
> In step S52, the ***BIOS image and the BIOS upgrade software*** corresponding to the model ID are read from the storage device (HDD) of the host computer (HOST), and are loaded into a memory. *In step S53, the corresponding BIOS image and BIOS upgrade software are transferred to the personal computer which requested the BIOS to be upgraded through the multiplexer (MUX) using serial communication.* In step S54, a transfer completion message, announcing that the ***transfer*** has been completed, is displayed after completing the transfer. Instep S55, a check is made as to whether the program terminate command of a user is input. If not, the process returns to step S50 to check whether the BIOS is requested to be upgraded. When the program terminate command of the user is input, the program is terminated.
>
> FIG. 6 is a flow chart showing a process for upgrading the BIOS executed in the personal computers of the respective terminals in a method for upgrading the BIOS using serial communication according to the present invention.
>
> Referring to FIG. 6, in the process of upgrading the BIOS executed in the personal computer which requested the BIOS to be upgraded, when the personal computer is booted in step S60, the power-on self-test (POST) of the BIOS is started in step S61. In step S62, a check is made as to whether the BIOS is connected to the host computer (HOST). If not, the process proceeds to step S70 and normal power-on self-test is continued. When it is determined in step S62 that the personal computer is connected to the host computer (HOST), the host computer (HOST) is requested to upgrade the BIOS in step S63. Then, in step 64, the model ID is transferred to the host computer (HOST).

> In step S65, *the corresponding **BIOS image and BIOS upgrade software** are received from the host computer (HOST) and are stored in the memory.* In step S66, *the BIOS upgrade software stored in the memory is driven.* In step S67, the existing BIOS stored in the BIOS ROM is removed by clearing the BIOS ROM. In step S68, **the BIOS image is written into the BIOS ROM**. In step S69, **the upgrade is completed by rebooting the computer**. (Col. 4, line 39 – Col. 5, line 25, emphasis added)

The flowchart of Fig. 5 corresponds to the transfer of a BIOS image and BIOS upgrade software from a host computer to a personal computer connected to the host computer via a serial communication scheme that includes a multiplexer. There is no load and execution of a second portion of platform firmware code on the personal computer during the process of Fig. 5.

The flowchart of Fig. 6 corresponds to a process for upgrading the BIOS for multiple personal computers using a serial communication scheme. During this process, a BIOS image and BIOS upgrade software are received from the host computer and loaded into memory for a personal computer for which an upgrade is being performed. "In step S66, the BIOS upgrade software stored in the memory is driven" (*i.e.*, executed). Execution of the BIOS upgrade software is used to update the BIOS for the personal computer by replacing an existing BIOS (image) with the new BIOS image (the BIOS image received from the host computer).

The BIOS upgrade software is not a portion of platform firmware (code). It is software that is not a part of a platforms BIOS or firmware code, as clearly recited, and is used for the purpose of upgrading BIOS (*i.e.*, firmware) stored in a ROM. Under *Ha*, there is no concept of loading a second portion of platform firmware code from a remote storage device and executing that platform firmware code as part of a platform firmware load process performed during a pre-boot phase of a computer system.

In addition, under *Ha* the BIOS is monolithic. That is, it comprises a single image, as is the conventional form of BIOS. The single BIOS image is stored in a

non-volatile memory device (*e.g.* ROM, Flash memory, *etc.*) on board the personal computer and loaded during the pre-boot process to perform operations typically associated with platform firmware, such as verifying system integrity and configuration and prepare the platform for booting an operating system (hence the "pre-boot" name of the phase prior to booting an operating system). To complete an upgrade under *Ha*, the personal computer must be rebooted. During the reboot process, the new BIOS image (which has now replaced the former image) is loaded and executed in its entirety during a second pre-boot phase. This clearly does not anticipate the claimed invention of amended claim 1.

In contrast, under embodiments of the present invention, such as recited in claim 1, a first portion of the platform firmware code is loaded from a local firmware storage device (*e.g.*, a PROM or flash chip) and executed, while a second portion of the platform firmware code is retrieved from a remote storage device and executed by the platform during the same pre-boot phase. Thus, the platform firmware comprises a first locally-stored portion and a second-remotely stored portion. Under such a scheme, the platform firmware can be effectively upgraded by simply changing the second (remotely-stored) portion of platform firmware code. Under the invention of claim 1, there is no need to replace or add to the first portion of platform firmware code that is locally stored to effect an update.

In view of the foregoing, it is clear that *Ha* cannot anticipate independent claim 1. Accordingly, the rejection of claim 1 is improper and should be withdrawn.

With respect to the rejection of independent claim 13 as anticipated by *Ha*, this claim is a method claim reciting operations similar to those recited in claim 1, with the further limitation of having the first portion of platform firmware code loading a driver that enables access to a firmware volume comprising a storage device in which a second portion of platform firmware code is stored. Accordingly,

independent claim 13 is clearly not anticipated by *Ha* for similar reasons to those presented above in support of the traversal of the anticipated rejection for claim 1.

With respect to the patentability of independent claim 17, this claim has been amended to now recite,

17. (Currently Amended) A method comprising:

updating platform firmware for a computer system during a system boot-up operation comprising a single pre-boot phase by executing an early portion of platform firmware code that is stored locally in the computer system, causing the computer system to perform the operations of:

retrieving an updated set of platform firmware code from a remote firmware storage device;

updating platform firmware code that is stored locally in the computer system by replacing at least a portion of existing platform firmware code with the updated set of platform firmware code or adding the updated set of platform firmware code thereto; and

executing a remaining portion of platform firmware code to complete the system boot-up operation during the single pre-boot phase, the remaining portion of platform firmware code including the updated set of firmware code. (Emphasis added)

The system boot-up operations correspond to load and execution of the platform firmware to initialize the platform (*i.e.*, system) and prepare the system for booting an operating system, as is well-known. It is analogous to the foregoing "pre-boot phase," which is now explicitly recited. As discussed above, the upgraded firmware image under *Ha* is not executed during the same pre-boot phase (system boot-up operations) as the first (early) portion of platform firmware code. Rather, an entire BIOS image is replaced with a new upgraded BIOS image, which is then

loaded and executed during a subsequent re-boot. Accordingly, it is clear that claim 17 is not anticipated by *Ha*.

With respect to independent claim 22, this claim recites a computer system for performing operations analogous to similar operations recited in method claim 1. Accordingly, claim 22 is patentable over *Ha* for similar reasons to those argued above in support of the patentability of claim 1.

With respect to independent claim 26, this claim recites a non-volatile memory component that stores a first portion of platform firmware instructions for performing operations analogous to similar operations recited in method claim 1. Accordingly, claim 22 is patentable over *Ha* for similar reasons to those argued above in support of the patentability of claim 1.

Argument in Support of the Patentability of the claims over *Ha* and *Rasmussen*

Currently, each of independent claim 1, 13, 17, 22, and 26 stands rejected as anticipated by *Ha*, while dependent claims 8, 10, 13, 14, 25 and 30 stand rejected as being unpatentable over *Ha* in view of *Rasmussen*. An argument is presented above in support of the patentability of the independent claims. However, in view of a potential future rejection of the independent claims over either *Rasmussen* or the combination of *Ha* and *Rasmussen*, the following argument is now presented.

Applicants respectfully assert that none of independent claims 1, 13, 17, 22, and 26 is anticipated by *Rasmussen* or obvious in view of either *Rasmussen* alone, or the combination of *Ha* and *Rasmussen*.

To establish a *prima facie* case of obviousness, there must first be some suggestion or motivation to modify a reference or to combine references, and second be a reasonable expectation of success. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art and not based on applicant's disclosure. Third, the prior art reference (or references when combined) must teach or suggest all the claim

limitations. M.P.E.P. § 706.02(j) from *In Re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991). Where claimed subject matter has been rejected as obvious in view of a combination of prior art references, a proper analysis under § 103 requires, *inter alia*, consideration of two factors: (1) whether the prior art would have suggested to those of ordinary skill in the art that they should make the claimed device; and (2) whether the prior art would also have revealed that in so making, those of ordinary skill would have a reasonable expectation of success. Both the suggestion and the reasonable expectation of success must be founded in the prior art, not in the Applicants' disclosure. *Amgen v. Chugai Pharmaceutical*, 927 F.2d 1200, 18 USPQ2d 1016 (Fed. Cir. 1991), *Fritsch v. Lin*, 21 USPQ2d 1731 (Bd. Pat. App. & Int'f 1991). An invention is non-obvious if the references fail not only to expressly disclose the claimed invention as a whole, but also to suggest to one of ordinary skill in the art modifications needed to meet all the claim limitations. *Litton Industrial Products, Inc. v. Solid State Systems Corp.*, 755 F.2d 158, 164, 225 USPQ 34, 38 (Fed. Cir. 1985).

The examiner must present a convincing line of reasoning as to why the artisan would have found the claimed invention to have been obvious in light of the teachings of the references. M.P.E.P. § 70602(j) from *Ex parte Clapp*, 227 USPQ 972, 973 (Bd. Pat. App. & Inter. 1985). Obviousness cannot be established by combining references without also providing evidence of the motivating force which would impel one skilled in the art to do what the patent applicant has done. M.P.E.P. § 2144 from *Ex parte Levengood*, 28 USPQ2d 1300, 1302 (Bd. Pat. App. & Inter. 1993) (emphasis added by M.P.E.P.).

Description of aspects of the operation of *Rasmussen's* "Method and Apparatus of Remotely Updating Firmware of a Communication Device" are contained in the SUMMARY OF THE INVENTION section. The following paragraph is illustrative of the method disclosed by *Rasmussen*:

> Accordingly, an aspect of the present invention provides a method of remotely updating firmware saved in a FLASH memory of a communication device connected to a network, the FLASH memory being partitioned into at least a first and a second portion, a first firmware load being stored in the first portion, the method comprising the steps of, during run-time of the first firmware load: transmitting at least a portion of an updated firmware load through the network to the communication device; temporarily storing the transmitted portion of the updated firmware load in a buffer of the communication device; copying at least a portion of the first firmware load into a random access memory (RAM) of the communication device; triggering execution of the copied portion of the first firmware load from the RAM; and under control of the copied portion of the first firmware load, saving the transmitted portion of the updated firmware load to the second portion of the FLASH memory. (Col. 3, lines 34-45)

It is noted that under no embodiment does *Rasmussen* execute the updated firmware load during the run-time of the first firmware load (*e.g.*, during a pre-boot process). Rather, a portion of the first firmware load is copied into RAM and executed to save the transmitted portion of the updated firmware load into a second portion of the FLASH memory. The portion of firmware comprising the updated firmware load is only executed during a subsequent reboot, as is clearly recited in the SUMMARY OF THE INVENTION section, as follows:

> In accordance with the present invention, after completion of the update procedure, the updated version of the firmware becomes active ***upon a subsequent reboot of the communication device***. In one embodiment, this is accomplished by the user during their normal use of the communication device: that is, by powering down and later re-booting their PC. In an alternate embodiment, communication device reboot can be accomplished automatically as a final step in the update procedure. (Col. 4, lines 36-44, emphasis added)

It is clear that *Rasmussen* does not anticipate any of independent claims 1, 13, 17, 22, or 26. Furthermore, it is clear that each of independent claims 1, 13, 17, 22 and 26 is clearly patentable over an obviousness rejection in view of *Ha* alone, *Rasmussen* alone, or the combination of *Ha* and *Rasmussen*, for at least the reason that each independent claim includes at least one element or limitation that is not

disclosed or fairly suggested in either of the *Ha* and *Rasmussen* references, thus failing the third prong of the *In Re Vaeck* test.

Conclusion

Overall, it is clear that none of the references, singly or in any motivated combination, disclose, teach, or suggest what is recited in these independent claims. Thus, given the above amendments and accompanying remarks, independent claims 1, 13, 17, 22 and 26 are now in condition for allowance. The dependent claims that depend directly or indirectly on these independent claims are likewise allowable based on at least the same reasons and based on the recitations contained in each dependent claim.

If the undersigned attorney has overlooked a teaching in any of the cited references that is relevant to the allowability of the claims, the Examiner is requested to specifically point out where such teaching may be found. Further, if there are any informalities or questions that can be addressed via telephone, the Examiner is encouraged to contact the undersigned attorney at (206) 292-8600.

Charge Deposit Account

Please charge our Deposit Account No. 02-2666 for any additional fee(s) that may be due in this matter, and please credit the same deposit account for any overpayment.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN

Date: Aug 22, 2005

R. Alan Burnett
Reg. No. 46,149

12400 Wilshire Boulevard
Seventh Floor
Los Angeles, CA 90025-1030